

XP-002283680



Authentication Reference Guide

P.D. 09-04-2002	18
P. 1-12	

September 2002

Disclaimer:

This white paper contains Secure Computing Corporation product performance information that is meant as a guide to customers. Performance is greatly influenced by traffic profiles and how the hardware and software are set up and configured. Therefore, the performance data within implies no guarantee by Secure Computing that the customer will realize the same performance levels.

**SECURE
COMPUTING**

**Secure Computing Corporation
Corporate Headquarters**
4810 Harwood Road
San Jose, CA 95124
tel +1.800.379.4944
tel +1.408.979.6100
fax +1.408.979.6501

International Headquarters
tel +44.1753.410900
fax +44.1753.410901

www.securecomputing.com

BEST AVAILABLE COPY

TABLE OF CONTENTS

Authentication Reference Guide

What's covered in this white paper	3
Introduction: acceptable and appropriate authentication	4
What is "acceptable and appropriate" authentication	4
Basic types of authentication	5
What type of authentication is best	5
Five important requirements for password	6
Weighing the types of authentication	6
No authentication	6
Authentication by memorized passwords	7
Authentication by PIN	8
Authentication by CHAP	9
Authentication by password authenticators or tokens	11
Authentication by smart cards	16
Authentication by user-based	
locally-stored digital certificates	16
Authentication by biometric traits	17
A word about cost	18
Summing up: When are memorized passwords ok?	18
For more information	18



Authentication Reference Guide

A guide to leading industry authentication methods-from passwords to digital certificates to biometrics.

Their pros and cons.

And deciding which method is best for you.

What's covered in this white paper

Computers have never been very good at identifying authorized users and separating them from outsiders. Modern computer networks and generally accepted business practices require that operators of computer systems and networks consider the sensitivity of the information contained within or processed by their systems, the risks of unauthorized access, and usually, appropriate means to establish user accountability.

In this paper you'll find a discussion of the requirement to adequately identify all system users, followed by a description of each of the available technical means of positively identifying users requesting access to sensitive or valuable information. Different methods with varying security levels are compared and contrasted. Particular attention is paid to the issues of compatibility, because some of the promising technology for identifying people simply won't work on a large scale until the world's computing infrastructure changes. This paper also suggests standards of generally acceptable practice with regard to positive identification of users of computer equipment or networks.



Introduction: acceptable and appropriate authentication

Modern information security systems are composed of three primary components which have come to be known as:

- Authentication,
- Authorization, and
- Accountability

Authentication is the most fundamental of these three elements, because it precedes the other two. The term "authentication" or "user authentication" refers to confirming the identity of a person requesting access to computers, networks, or computerized resources. *Authorization* rules restricting a person's activities have little value if that person cannot be distinguished from another who is not restricted. And *accountability* or audit records cannot deter users from abusing their authority if the credibility of the audit's association with specific user activities can be challenged. Even systems that encrypt all information are of limited value if they decrypt information at the request of both identified and unidentified users. What this means is that *acceptable and appropriate authentication must be in place before authorization rules, encryption systems, or audit mechanisms are effective!*

What is "acceptable and appropriate" authentication?

When deploying any security system, there are often trade-offs between security and convenience. Organizations must balance the value of the information being protected with ease-of-use and cost issues to deploy a system that is appropriate for their needs. The reality is that a system that is "secure enough" and used by a large number of people, may be more valuable than a highly secure system that is only implemented by a few. Because organizations have different levels of risk, multiple solutions are needed with varying levels of security strength.

As a reminder to our customers, the following guidelines should be observed when selecting, expanding, and implementing their user authentication system:

- 1) Perform an assessment to determine the value of what's at risk, and select an authentication method that is appropriate. In general this will be fixed passwords for low value data, software authenticators for medium value data, and hardware authenticators for medium to high value data.
- 2) When deploying software-based authenticators, educate your user community about safeguarding the device and any associated data that might contain private keys.
- 3) Use attack lockout features to automatically disable an account if it is under repeated attack.
- 4) Conduct periodic assessments of the value of the data at risk. If the risk has increased, take steps to implement a stronger authentication system.

By following these guidelines, organizations can deploy the user authentication security system that is appropriate for their needs, and not pay for more than they actually need.

There is actually quite a range of authentication methods, and Secure Computing offers multiple levels of security for user authentication. Some of these are more sophisticated, compatible, or expensive than others. Putting



Authentication Reference Guide

"acceptable and appropriate" authentication in place means striking a balance between the expense and complexity of the authentication method, the sensitivity of the information or assets being protected, and the vulnerabilities of and compatibility with the environment(s) from which authentication is accessible.

Here's what you need to consider in order to achieve an acceptable balance:

- *Sensitivity of information:* Is the most sensitive information accessible from the computer classified, public-domain, commercially valuable, or sensitive but unclassified government data?
- *Physical access:* Are all people who can physically enter any areas where the information or protected resources are available authorized to access all of it?
- *User accountability:* Are users held accountable for their actions? Businesses or agencies that pay internal or outside auditors to publish financial or technical reviews for shareholders or to meet legal obligations generally need to hold their computer users accountable for their work. Thus, user accountability corresponds with the need to audit user activities. User accountability is needed, for example, wherever there is a risk of unauthorized access to information, compromising the privacy of a citizen (for government computers), damaging the property of others, obtaining illegal personal gain, etc.
- *Compatibility:* Is the authentication method compatible with the computing environment? A system that is incompatible with important processes, applications, computers, or network equipment will not be used no matter how well it confirms user identity. Some popular authentication systems such as memorized passwords and RADIUS have achieved near-universal compatibility. Others, such as smart-card readers and biometrics, cannot be widely deployed until the world's computing and networking infrastructure changes.

Some authentication technologies work well for identifying employees or partners where a well-defined personal relationship is already established. Others work better for establishing enough initial identity to set up a trust relationship among former strangers, such as customers wishing to purchase goods over the Internet.

Basic types of authentication

There are four basic ways for computer systems to authenticate users. The computer can ask for something only the authorized user knows, test for the presence of something only the authorized user has, obtain some non-forgeable biological or behavioral measurement of the user, or determine that the user is located at a place where only the authorized user can enter. These different methods can be used in combination to increase the level of security, to adapt to special needs, or to compensate for identified vulnerabilities or incompatibilities.

What type of authentication is best?

After considering the four basic factors covered above with regard to sensitivity of information, physical access, user accountability, and compatibility, computer security plans and systems can determine the type(s) of authentication needed. The best solutions usually involve a mixture of two or more different authentication methods, harmoniously managed. Informed personal judgments will need to be used in many cases in order to make sensible and appropriate determinations balancing cost and complexity with trust, compatibility, and dependability. In other cases, the standards of generally accepted practice or the unyielding demands of compatibility will make the choices obvious. For example, computer or network systems that require authorization or audit also require strong, compatible authentication.



Five important requirements for passwords

In today's environment, wherever user accountability is required, authentication by memorized passwords alone is acceptable only if all five of the following conditions are met:

1. The password is known only to the corresponding user and is recorded nowhere—or at least, is not available in unencrypted form.
2. The environment does not allow installation of equipment capable of storing the password and replaying it without requiring the user to enter it from his or her memory.
3. Routine use of the password does not occur in areas where other users are in close proximity.
4. The same password is not used in any other computer or environment not meeting all five of these requirements. (For instance, does the user have the same password on several accounts?)
5. The delivery means by which the password is transmitted from the user to the ultimate destination(s) does not expose it to any other person, computer or environment not meeting all five of these requirements. (Note that most networks don't meet this requirement, at least not without encryption.)

Few computers and networks can meet all five of these requirements. Usually a memorized password system can be used to meet some of the requirements, and additional means can be added to compensate for the extra vulnerabilities associated with failure to comply with one or more of the above requirements. For example, failure to comply with requirement 5 above can usually be overcome by careful application of encryption to the password in transit (but this must be done carefully: if it is always encrypted in the same way, the encrypted representation will probably still be vulnerable to simple capture and replay on networks.)

Throughout the remainder of this paper, we will make frequent reference to the five requirements above, because they are well understood and because many modern authentication systems include a memorized password component as part of their overall toolset.

Weighing the types of authentication

No authentication

Today's generally accepted business and computing practices do not require user authentication *if*:

- All of the information accessible through or contained within a computer or network system is available in the public domain; or
- The environment containing all points from which the information or resources can be obtained is physically secured at all times so that only persons with authorized access can enter, *and* if there is no reason for user accountability.

In such cases, it is commonplace for computers and networks to operate with no automated attempt to identify users requesting access.



Authentication Reference Guide

Using no authentication

Advantages of using no authentication:	Easy to implement, no cost.
Disadvantages:	Depends for security on user accountability.
Security rating:	Very poor.
Compatibility rating:	Very good.
Ease of use/management:	Very good.

Authentication by memorized passwords

In the past, almost all computer security systems attempted user authentication by memorized passwords alone. Virtually all computers and networks are compatible with memorized password authentication schemes. However, powerful, worldwide computing trends during the past two decades have made them ineffective for security in most situations.

Studies by the U.S. Federal Bureau of Investigation and the Congressional hearings that led to passage of the U.S. Computer Security Act of 1987 revealed that most computer crimes have resulted from the inadequacies of memorized password systems. Vulnerable areas include personal computers (users program their PCs to store their passwords and automatically replay them at the touch of a keystroke or at the click of a mouse whenever requested) and local area networks, where passwords are broadcast to all stations and can be captured and replayed unless special steps are taken to encrypt them in sophisticated and dynamic ways.

In the past, the prescribed remedy was to assign passwords of longer length and greater complexity, and to demand that users change and memorize new ones at ever-increasing frequencies. Experience has clearly shown that these two methods do not work in modern environments, and obviously cannot help in the two vulnerable areas mentioned in the preceding paragraph.

With the advent of public networks, private networks based on broadcast topologies, personal computers, and computer-literate work groups, generally accepted practice requires that the use of memorized passwords be reconsidered. Much stronger methods of authentication may be required in many situations.

Using authentication by memorized passwords

Password is:	Something only the user knows.
Advantages of using memorized passwords:	Compatible with all systems. Easy to remember in many cases.
Disadvantages:	Very subject to abuse and interception. Cannot safely be sent across public network.
Security rating:	Poor.
Compatibility rating:	Good.
Ease of use/management:	Good.



Authentication Reference Guide

How memorized passwords measure up

1. Password known only to corresponding user and recorded nowhere in the clear?	⊖
2. Environment does not allow storing and replaying password?	⊗
3. Password is not subject to theft by other users in close proximity?	⊗
4. Password guaranteed not to be used on other computers?	⊗
5. Password is not exposed during transmission?	⊗

A word about memorized passwords

Though less secure, under certain low-risk circumstances passwords may be your preferred option for convenience. An example of this would be employees logging into a secure company network on trusted company premises. Low-level security here may be adequate and time-saving for employees. Memorized passwords can also function as a temporary fallback authentication mechanism if an individual loses a physical smart card or token.

Many of the minimum- and medium-security solutions work for companies who value ease-of-use and convenience. Any of these solutions can be made stronger or weaker depending on your company's security policy, as well as the implementation of the authenticators or the addition of another security factor (alternative factors are described below).

While memorized passwords are more convenient for users, they are not necessarily easier to administer and manage. Numbers of employees, their locations, and their roles are constantly changing, which affect password usage daily. Also, many users may have multiple passwords, depending on the number of applications they use. SafeWord provides advanced, streamlined management features and tools for efficiently handling such circumstances.

Authentication by PIN

A more sophisticated variation on the memorized password is the Personal Identification Number, or "PIN," as defined by the American Banker's Association. PINs are widely used in combination with bankcards and automated teller machines.

Some people think a PIN is simply a memorized password comprising numbers only. However, the PIN is usually encrypted and/or mathematically combined with some dynamically changing value that is known to both the sender and the receiver, and that is compensated for at both ends. The characteristic that separates a PIN from a password is that it can be transmitted across public networks without any compromise, even if adversaries are capable of monitoring, recording, and replaying network traffic.

Because the mechanism that protects the PIN must be complex enough to withstand attack, PIN systems usually require extra hardware at the location of the user, at the location of the protected computer, or both. Compatibility of this extra hardware with existing equipment usually requires careful planning and, sometimes, considerable expense. PINs usually rely on shared equipment that is not dedicated to an individual user. (Imagine how expensive banking would become if the bank had to build a separate ATM machine for each customer!) When implemented in accordance with ABA standards, PINs are acceptable for authentication wherever



Authentication Reference Guide

memorized passwords are acceptable, and also in environments where memorized passwords fail to meet requirements 2 and 5 regarding equipment capable of password storage and replay, and network exposure.

Using authentication by PIN

Password is:	Something only the user knows.
Advantages of using PINs:	<ul style="list-style-type: none"> Easy to remember yet fairly secure, especially when used in conjunction with ATM card. Can be transmitted safely across public networks.
Disadvantages:	<ul style="list-style-type: none"> Requires dedicated hardware and, usually, a card system. No compatibility across different systems.
Security rating:	Fair.
Compatibility rating:	Poor.
Ease of use/management:	Fair.

How PINs measure up

1. Password known only to corresponding user and recorded nowhere in the clear?	☹
2. Environment does not allow storing and replaying password?	☹
3. Password is not subject to theft by other users in close proximity?	☹
4. Password guaranteed not to be used on other computers?	☹
5. Password is not exposed during transmission?	☹

Developers of dynamic password systems (discussed below) have found very effective ways of incorporating PIN technology into their devices while solving the compatibility issues, all at relatively low cost.

Authentication by CHAP

In response to the need for a stronger memorized password authentication system—one that is appropriate for use in public networks—the Internet Engineering Task Force has published standards and guidelines describing a protocol known as “CHAP” (Challenge Handshake Authentication Protocol). Using this protocol, appropriately designed applications and networking equipment can issue cryptographic challenge/response conversations to establish their identities to one another.

CHAP authentication is usually automatic and transparent to users. In fact, CHAP is oriented more toward helping “black boxes” communicate than toward user authentication. CHAP is commonplace in modern network gateway devices, such as routers and commservers, which demand and evaluate CHAP-encoded memorized passwords (originating, for example, in PCs running Microsoft Windows software) before granting Internet connectivity.

CHAP authentication is compatible with virtually all router and commserver equipment and can therefore be installed at almost any Internet gateway. It is also compatible with most PPP client software, including the very



W H I T E P A P E R

Authentication Reference Guide

popular PPP clients that are distributed with Microsoft Windows. It is not, however, compatible with most "legacy" applications, including most mainframe and minicomputer login systems.

CHAP is strong enough to resist attack when sent over the Internet. However, when CHAP is fully automated and transparent it doesn't really confirm the identity of the human user. (Anyone who can gain access to the PC can switch it on and the router will let them on the network.) Moreover, even if a memorized password is demanded and CHAP-encoded, it is still subject to "shoulder surfing," where an unauthorized user watches the fingers of someone entering a password. Accordingly, generally accepted computer practices permit use of CHAP authentication in any situation where memorized passwords are acceptable, and also in situations where requirement 5 (network exposure) makes memorized passwords unsafe when used alone. However, even the best implementations of CHAP cannot solve the problems associated with requirement 3, where routine use of the memorized password occurs in close proximity with other computer-literate users.

Using authentication by CHAP

Password is:	Something only the user knows.
Advantages of using CHAP:	Good for sending over Internet. Compatible with router and commserver equipment. Can therefore be installed at almost any Internet gateway. Also compatible with most PPP client software.
Disadvantages:	Doesn't confirm identity of the human user. Not compatible with most mainframe and minicomputer login systems.
Security rating:	Fair.
Compatibility rating:	Good.
Ease of use/management:	Fair.

How CHAP measures up

1. Password known only to corresponding user and recorded nowhere in the clear?	☺
2. Environment does not allow storing and replaying password?	☺
3. Password is not subject to theft by other users in close proximity?	☹
4. Password guaranteed not to be used on other computers?	☺
5. Password is not exposed during transmission?	☺



© 09/04/02. SCC090402. Secure Computing Corporation. All Rights Reserved. SafeWord, PremierAccess, Sidewinder, Type Enforcement, SecureOS, Softoken II, Strikeback, SmartFilter, Power-It-On!, Safe, secure extranets for e-business, Access control without limits, and Plug into a positive Web Experience are either registered trademarks or trademarks of Secure Computing Corporation. All other trademarks, tradenames, service marks, service names and images mentioned and / or used herein belong to their respective owners.

Authentication by password authenticators or tokens

An "authenticator" or a "token" is a device used to generate dynamic passwords while logging onto a network. Token types range from software tokens (that may reside on PCs, PDAs, cellular phones, smart cards, etc.) to hardware tokens that resemble small hand-held calculators.

Secure Computing's tokens calculate dynamic passwords that work only once. In some cases as you will read below, not all dynamic passwords are one-time passwords, and this is an important distinction. All our tokens use "dynamic password technology" to generate true one-time passwords for a higher level of security. The beauty of such one-time passwords is that any password that may be captured by a hacker (for instance by shoulder surfing) after it has been used will no longer be authenticated by the network because it is no longer valid.

To add an additional layer of security, users can type in a PIN with the password when logging in to the network (often called a SoftPIN because it's software-based). Tokens such as Secure Computing's SafeWord Platinum require the user to enter a PIN into the token itself in order to activate it. This token provides the highest level of security and is our recommended method when security needs are critical.

Basically, tokens may use two methods of authentication: synchronous and asynchronous (challenge-response).

A. Synchronous (sync) authentication

Synchronous dynamic password systems are the easiest tokens for users to operate.

There are two types of synchronous methods: time-synchronous and event-synchronous. (Secure Computing supports event-synchronous tokens only.) Both methods, and their characteristics, are discussed and compared below.

1. Event-synchronous authentication

As stated above, tokens using event synchronization are very easy to use. Simply push a button and a new dynamic password is displayed. The cryptalgorithm incorporated in the token uses a counter that stays "in sync" with the server based on the number of passwords generated.

Some of the advantages of event-synchronous authentication are as follows. First, because these tokens rely on the synchronizing of an event between the token and the server (which is the generation of the dynamic password), the problems that can be encountered by time synchronization are not factors. (See below for a description of potential problems with time synchronization.)

The really good news about event-synchronous tokens, like those from Secure Computing, is that the dynamic password is completely secure because it truly can only be used once. These "one-time passwords" create an impenetrable authentication solution for your extranet-even if the password is stolen, it can't be used to break in. This delivers a much higher level of security than memorized passwords (which can be easily stolen) or time-synchronous tokens (which can be used more than once within the several minutes that the password is valid.



Authentication Reference Guide

Using a event-synchronous authenticator

Key is:	Something only the user has.
Advantages of using a event-synchronous authenticator:	<p>Gives much more security than simple memorized password.</p> <p>Better security and easier administration than time-dependent authentication.</p> <p>Provides best protection against many common attacks.</p> <p>Gives excellent protection against shoulder surfing.</p> <p>No card reader required.</p> <p>Is not vulnerable to replay attacks.</p>
Disadvantages:	<p>Administrative overhead is greater than memorized passwords.</p> <p>Authenticator is subject to loss or theft.</p>
Security rating:	Very good.
Compatibility rating:	Good.
Ease of use/management:	Good.

How event-synchronous authenticators measure up

1. Password known only to corresponding user and recorded nowhere in the clear?	☺
2. Environment does not allow storing and replaying password?	☺
3. Password is not subject to theft by other users in close proximity?	☺
4. Password guaranteed not to be used on other computers?	☺
5. Password is not exposed during transmission?	☺

2. Time-synchronous authentication

Time-synchronous authenticators also generate unique, dynamic passwords at fixed intervals, usually one per minute. They are also extremely easy to use because the tokens are always ON, generating a different password continually. Secret keys and an encryption scheme ensure that the passwords generated are unique to the authenticator.

Because the time continuously changes, in principle, each authentication attempt uses a unique password. But there is a risk that this password could be used more than once. Here's how. The server and the token must remain in tight sync with each other. For this to occur, the token must be ON at all times. Because all clocks are subject to drift, periodic resynchronization of the token and server is necessary. Often, to prevent drifting from interrupting the synchronization, and from interfering with authentication, this tight synchronization may be relaxed slightly. This is done by specifying a small period in time (up to seven minutes) during which the dynamic password is valid. The server will then accept the password as many times as it is used during this window of time where the password is valid, making it possible for more than one user to utilize the password for a



Authentication Reference Guide

brief period. This factor, combined with the fact that the token is ON all the time, creates a possible window of opportunity for someone to see another user's password by "shoulder surfing." That is, another user might see a password on the token at a glance, giving that unauthorized user several minutes to use the password on the network and log in as the rightful user. This could happen even if the token itself requires a PIN to be activated.

As a deterrent to this, many networks that use time-synchronous tokens require a SoftPIN. The problem with this is that another person on the network can view the password+PIN combination during transmission, and thus capture the PIN-and the password, which may still be good for several minutes.

Other potential problems exist as well. For instance, if the clock on the server or the token is reset to a different time zone, the passwords generated by the token are no longer valid. This could happen when a mobile user attempts to connect into a large corporate network spanning several time zones, with subnets in each time zone using local authentication servers. The clock corresponding to the local authentication server may belong to a different time zone and is, therefore, not synchronized with the user's token. In this case, although the user has a registered token, authentication fails. There are ways to overcome this problem, but the solutions often require considerable administrative overhead. One workaround to this problem is to synchronize the token and the server using Greenwich Mean Time (GMT).

Using a time-synchronous authenticator	
Key is:	Something only the user has.
Advantages of using a time-synchronous authenticator:	<ul style="list-style-type: none"> Gives much more security than simple memorized password. Provides good protection against many common attacks. No card reader required.
Disadvantages:	<ul style="list-style-type: none"> Administrative overhead is greater than memorized passwords. Authenticator is subject to loss or theft. Does not provide protection against shoulder surfing. Is vulnerable to replay attacks. Requires more administration than event-dependent authentication.
Security rating:	Good.
Compatibility rating:	Good.
Ease of use/management:	Fair.

How time-synchronous authenticators measure up	
1. Password known only to corresponding user and recorded nowhere in the clear?	☺
2. Environment does not allow storing and replaying password?	☺
3. Password is not subject to theft by other users in close proximity?	☹
4. Password guaranteed not to be used on other computers?	☺
5. Password is not exposed during transmission?	☺



Authentication Reference Guide

B. Asynchronous (challenge-response) authentication

Asynchronous authenticators were developed and marketed before other dynamic password systems. They are easy to understand and provide the best protection against well-known attacks. In an asynchronous scheme, the authentication procedure follows this script:

1. The authorized user attempts to log into the system.
2. The system presents the user with a short, random number (a challenge).
3. The user enters this random number challenge into the keyboard of the authenticator, which provides a response.
4. The user sends the response back to the system.

Such challenge/response authentication is simple and intuitive, and the challenge and response are not connected to any other event. (Like synchronous tokens, asynchronous tokens may also require a PIN for usage.)

Using an asynchronous authenticator	
Key is:	Something only the user has.
Advantages of using an asynchronous authenticator:	<p>Gives much more security than simple memorized password.</p> <p>Provide best protection against common attacks, including replay attacks.</p> <p>Gives excellent protection against shoulder surfing.</p> <p>Simple to administer compared to other dynamic password methods.</p> <p>No card reader required.</p>
Disadvantages:	<p>Administrative overhead is greater than memorized passwords.</p> <p>Authenticator is subject to loss or theft.</p> <p>Usability is not as good; requires user to enter a challenge into the authenticator, then copy the response into the system.</p>
Security rating:	Very good.
Compatibility rating:	Fair.
Ease of use/management:	Fair.

How asynchronous authenticators measure up	
1. Password known only to corresponding user and recorded nowhere in the clear?	☺
2. Environment does not allow storing and replaying password?	☺
3. Password is not subject to theft by other users in close proximity?	☺
4. Password guaranteed not to be used on other computers?	☺
5. Password is not exposed during transmission?	☺



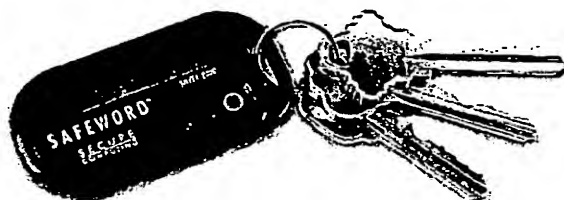
© 09/04/02. SCC090402. Secure Computing Corporation. All Rights Reserved. SafeWord, PremierAccess, Sidewinder, Type Enforcement, SecureOS, Softoken II, Strikeback, SmartFilter, Power-It-On!, Safe, secure extranets for e-business, Access control without limits, and Plug into a positive Web Experience are either registered trademarks or trademarks of Secure Computing Corporation. All other trademarks, tradenames, service marks, service names and images mentioned and / or used herein belong to their respective owners.

Authentication Reference Guide

Types of authenticators

Secure Computing offers examples of several types of authenticators. For example:

- **SofToken II:** Secure Computing's software-based token, SofToken II, is for users who want the security of one-time passwords but do not want to carry a hardware token. SofToken generates a dynamic password just as the handheld tokens do, but the SofToken software resides on the hard drive of the user's laptop or desktop system. [The dynamic password capability offers higher security than memorized passwords. All software-based methods of security are vulnerable if an attacker should gain access to the device data is stored on, and therefore the security level is moderate. But given this understanding, many of our customers select this solution for particular environments where this security provision satisfies their requirements.] SafeWord software tokens can also be run on Palm handheld computers and the Ericsson R380s smartphone. The token can be configured for synchronous or asynchronous authentication.



- **SafeWord Silver 2000:** In a convenient keyring package, SafeWord Silver 2000 authenticators generate one-time passwords with the simple touch of a button. These passwords can only be used once, ensuring secure access. No PIN is required with this authenticator. Synchronous only.
- **SafeWord Platinum:** This is the strongest authenticator for SafeWord, providing two-factor authentication, which is the combination of:
 - What a user knows (a PIN)
 - What a user has (an authenticator for generating one-time passwords)

A user first activates the token with a personal identification number (PIN), which prompts the token to provide a one-time, dynamic password. The user then enters the password into the network to perform authentication. The PIN is not vulnerable to theft or discovery through "sniffing" software, further ensuring security. If the token is stolen, the thief cannot retrieve valid passwords from the token without the PIN; what's more, as soon as the token is reported stolen, the administrator can immediately disable it. This type of token can be configured for synchronous or asynchronous authentication.



Authentication Reference Guide

Authentication by smart cards

A "smart card" is a small, credit-card-sized card. This card actually has built-in intelligence, thanks to an on-board chip. In the authentication application, the smart card contains the secret authentication information.

Smart cards are not yet compatible with the many types of personal computers and network workstations upon which modern commerce must rely. They are only truly "smart enough" when combined with an electronic reader and its associated power supply. The reader connects into a port, slot, or socket that then hooks into a networked computer.

For specialized, high-value or high-risk applications that are used by small, well-defined user groups, smart cards make sense; smart-card readers can be purchased, installed, configured, and accompanied by custom software applications. Unfortunately, adding the card reader makes this solution too expensive to deploy in large numbers. Nobody has yet figured out how to make a smart-card reader that's inexpensive enough to give one to every computer user. And computer users are reluctant to get out of their office chairs several times per day to stroll over to a shared departmental smart-card reader, even if it does keep the company's costs down.

Using a smart card	
Key is:	Something only the user has.
Advantages of Using a Smart Card:	Gives more security than simple memorized password.
Disadvantages:	Expensive. Compatibility is a problem. Primarily used in small, customized systems.
Security Rating:	Good
Compatibility Rating:	Poor
Ease of Use/Management:	Poor

How smart cards measure up	
1. Password known only to corresponding user and recorded nowhere in the clear?	☺
2. Environment does not allow storing and replaying password?	☺
3. Password is not subject to theft by other users in close proximity?	☺
4. Password guaranteed not to be used on other computers?	☺
5. Password is not exposed during transmission?	☺

Authentication by user-based locally-stored digital certificates

One way around the card reader conundrum is to take the secret information out of the smart card and hide it on the computer's disk drive somewhere, surrounded by software that mimics a smart card reader and tricks remote applications into thinking it is conversing with a real smart card through a real reader. However, security experts point out that this solution is not secure enough to establish user accountability, because the secret information could be seen by computer-literate workers.



Authentication by biometric traits

In recent years, equipment of various types, capable of accurately measuring fingerprints, retina patterns, hand geometries, hand motions while writing one's legal signature, or hand motions while typing on a computer keyboard, have been developed. The information carried by this equipment is generally referred to as biometric information, and it can usually be reduced to a recognizable, stable template for each individual user.

If the biological measuring equipment or software used in a given authentication system is able to reliably obtain unique templates for all of users, generally accepted security practice allows authentication via these biometric systems with a memorized password system in any situation where memorized passwords would be acceptable. These systems are also good for situations where memorized passwords cannot be safely used alone due to failure to meet requirement 3 regarding the physical security and accessibility of the environment.

But biometric traits are not a panacea: Because authorized users cannot change their biological signatures, biometric authentication systems must be carefully designed so as not to expose biological signatures to unsecure environments. This is because once a user's fingerprint, for example, has been digitized and then broadcast or transmitted to areas or systems where abuse or replay cannot be controlled, the corresponding user can never use prints of that finger for authentication again.

This has given rise to the notion of biometric systems that never actually transmit these biological signatures in a way that could be replayed, or that never reveal their actual value. The methods used to protect biological information vary in their implementation and level of sophistication. In the future, perhaps dynamic password authenticators such as those discussed above will demand some kind of biometric input before they will issue correct dynamic passwords.

Authentication by a properly implemented dynamic biometric authentication system is acceptable in any type of environment. But here's an important point: whether such biometric systems are static or dynamic, *the method used to protect the biometric signature must be stronger than the methods used to protect passwords or cryptographic keys because it must resist attack for the lifetime of the authorized user.* What's more, not only may the user be exposed to information of much greater value in the distant future, but there is real concern that it may not be possible for today's biometric products to encrypt biometric information with a cryptoalgorithm whose strength can be guaranteed for the lifetime of the corresponding person!

Compatibility is a big issue with biometric systems. Biometric equipment must be installed in or near each user's workstation, and network gateways (such as routers and commservers), login procedures, and sensitive applications must be reprogrammed to query this equipment. Because of the expense of most of this equipment, these tend to be small and highly customized systems.



Authentication Reference Guide

Using a biometric authenticator

Key is:	Non-forgeable biological or behavioral measurement of the user.
Advantages of using a biometric authenticator:	<p>Gives more security than simple memorized password.</p> <p>Provides protection against many common attacks.</p> <p>Gives excellent protection against shoulder surfing.</p>
Disadvantages:	<p>High administrative overhead.</p> <p>Small, highly customized systems only.</p> <p>Danger of leaving biometric password (fingerprint, voice print) everywhere.</p> <p>Even an encrypted biometric password may be "cracked" 10, 20, 30 years in the future, which would render that password useless.</p>
Security rating:	Fair.
Compatibility rating:	Poor.
Ease of use/management:	Poor.

How biometric authenticators measure up

1. Password known only to corresponding user and recorded nowhere in the clear?	☹
2. Environment does not allow storing and replaying password?	☹
3. Password is not subject to theft by other users in close proximity?	☹
4. Password guaranteed not to be used on other computers?	☹
5. Password is not exposed during transmission?	☹

A word about cost

The cost of authentication systems meeting the requirements of generally accepted computing practices is significant enough to require careful consideration during the selection and implementation stages, and to require careful management when they are used. Costs are associated with acquisition, installation, customization, administration, user training, productivity impact, and maintenance of such systems. It is important that planners ask vendors for detailed statements, policies, and prices for each of these factors, and that planners determine the real cost of these systems rather than concentrate on one or two areas.

Summing up: When are memorized passwords ok?

In many situations memorized passwords are an adequate solution. But if you're paying auditors to audit you, you need accountability. And that means something more than memorized passwords.

For more information

Visit our Web site at:

www.securecomputing.com



© 09/04/02. SCC090402. Secure Computing Corporation. All Rights Reserved. SafeWord, PremierAccess, Sidewinder, Type Enforcement, SecureOS, SoftToken II, Strikeback, SmartFilter, Power-It-On!, Safe, secure extranet for e-business, Access control without limits, and Plug into a positive Web Experience are either registered trademarks or trademarks of Secure Computing Corporation. All other trademarks, tradenames, service marks, service names and images mentioned and / or used herein belong to their respective owners.

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.